

CONSISTENT USER INTERFACE FRONT END FOR REMOTE USER INTERFACES

The present invention relates to a system and method for a consistent front end for remote and local user interfaces.

Throughout the following text the Glossary and Abbreviations at the end of the description are employed, usually without reference to same. It is to be understood that these terms are used in the sense defined herein if there is a conflict between the ordinary meaning of these terms and the terms defined herein.

User interfaces are playing an increasingly important role in home networking. In the area of universal plug-and-play (UPnP), however, almost no standards have been either developed or proposed concerning user interfaces. Developers of user interface (UI) devices, consequently have complete freedom to create applications and user interfaces for controlling services in a home network. Controlled devices, such as an audio device or a DVD+RW device, currently can optionally provide an HTML-page as a user interface. Except using optional HTML-pages they have, in fact, very limited control over the appearance of a user interface that is used on a UI device and can never be sure that their distinguishing features are accessible from these UI devices. Furthermore, some home networking scenarios, such as session migration and the use of multiple output devices simultaneously, are difficult to realize if the ownership of applications and user interfaces belong solely to UI devices.

Remote UI technology is becoming an important technology in this area (next to local UIs) in order to deal with these limitations. PC manufacturers use it to allow applications running on a PC to be used on thin-client devices. Remote UI technology also enables a controlled device, such as a DVD+RW, to offer applications and UIs together with its services to other devices in a home network. This provides a way for developers to maintain ownership of user interfaces and applications since, it is important for participants to be able to distinguish themselves from others. Remote UI technology may also reduce the costs for upgrading UI devices. Therefore, it is likely that the various developers will want to have

control over both applications and their user interfaces, and that multiple scenarios of remote UI ownership will co-exist, driven by the marketplace for this technology.

Dealing with user interfaces in a home network is a very complex issue. Because of the large number of developers participating in this technology, very flexible solutions and more open devices are needed in order to meet all their requirements, such as providing a good out-of-the-box experience, being able to handle all kinds of (future) devices, services, applications, different configurations, and multi-user scenarios.

The pair of usage scenarios contained in Appendix A has been promulgated as a basis for developing UPnP remote I/O standards and is typical of home network usage. Considering these usage scenarios as a starting point, the following requirements are important for user interfaces in a home network.

- It must be possible to make use of the functionality that is offered by the services, devices, and applications within a home network via UI devices, preferably using a UI with a developer-specific look & feel. Generic UI devices must provide support for the majority of services/devices/applications that are available in a home network.
- UI devices can have varying capabilities, such as screen size, I/O modalities, etc. User interfaces must be tailored as much as possible to the specifics of the UI device, instead of a free-floating cursor (like in mouse-based systems) on a TV using a standard remote control, and not requiring extensive use of scrolling on a small-screen display. Furthermore, devices can be very resource constrained.
- In order to be successful, home network products must not be more difficult to use than non-networked products. UIs must automatically deal with network configuration changes. In addition, it is important that UIs be consistent (for similar controlled devices as well as among UI devices from different vendors). This means, for example, that:
 - a) preferably, all user interfaces on a generic UI device have the same look & feel, for example, changing volume is done with the same UI for every device and not by having a completely different UI for every developer (e.g. a slider with

label "Volume" in one case and a rotating knob with label "Vol. Level" in another case), and

- b) the interaction with a particular device/service/application must be similar among all the UI devices (e.g. not show "Play from DVR" on one UI device, "Play from HD" on a another device and "Play MPEG-2 files" on a PC, because this might create confusion and annoyance among users.
 - The cost for adding networking capabilities to devices must be kept at a minimum.

In addition to the scenarios of Appendix A, the following scenarios are important scenarios that networked user interface technology must be able to support:

- a) Access to content and control of devices: It must be possible for the user to have access to the content that is available in the network as well as the ability to control devices and services, from any compatible device anywhere inside the home, and in a coherent way. It must be possible, for example, to browse/search through all the content as if it were a single database (at least per category, such as through all MP3s), and render the desired content on a user-specified compatible rendering target. Besides content-browsing and device control applications, networked user interfaces must also be able to support communication-based applications, e-commerce, information services, games, productivity applications, and so on.
- b) Out-of-the-box experience: It must be easy for the user to add new devices to the home network. Example: User X has just purchased a generic UI device to control his home network. His home network consists of several UPnP devices. After switching on the control device it shows him that it has successfully detected several CE devices and that it has automatically downloaded the newest drivers for most of them. Unfortunately, one of them is not supported. The control device support center has been contacted about the problem and the user will be notified automatically when an update is available. In the meantime, the user is advised to download and install some bridging software

from the control device support center to the user's PC, with which the user can create simple HTML pages from service descriptions for use in the mean-time.

- c) Multiple users: Typically, a home network is used by more than one person. Different family members may want to use the same devices/services/applications of an in-home network at the same time. Therefore, it must be easy to switch to an alternative service or service instance, or to another UI device. It is desirable to share the same resource (e.g. an application) with other people. For example, it must be possible to play a game against each other, where each player has their own UI device and their own view of the same application. Synchronization is very important in such applications in a home network.
- d) Personalization: The settings of one user are probably not applicable to other users (e.g. musical preferences in an mp3-jukebox application). Furthermore, there is a need for communication among users. For example, send a "dinner is ready" message to everybody using the home network.
- e) Multiple I/O devices: It must be possible to use multiple input and multiple output (i.e. multiple UI) devices for the same application at the same time, to enable UIs where the graphical part of the UI is shown on the TV and the audible part of the UI is redirected to the speakers that are closest to the user. Or, to enable applications such as an intelligent photo browser with which the user can redirect pictures from a mobile device to a bigger screen.

Thus, there is a need for consistency of user interfaces across consumer electronic (CE) devices that are resident in a home network.

The present invention provides a system and method that uses an extensible database of "synonyms" containing constructs of a UI that can be used to replace correspondingly similar UI constructs (also called components) of a CE device, such as labels of UI elements and common user interface patterns, wherein a CE device can make incoming remote UIs more consistent with one another. In a preferred embodiment of a home network, one CE device is typically a UI control device for all the CE devices in the network. On power-up,

each of the CE devices sends its UI to the control CE device and it is replaced with a consistent UI, according to a preferred embodiment, whenever the control device presents that UI to a user.

The database of “synonyms” is used by the control CE device by examining each incoming remote UI for constructs and matching these found UI constructs with known similar “synonyms” in the database. Optionally, those found UI constructs that are not already in the database can be added to the database during this process (by the CE device locally and by downloading the found constructs to a CE device service center). A basic set of these “synonyms” is provided to initiate this system and method.

Note: the above described usage of a database of “synonyms” is not to be confused with using synonyms in speech where the intent is to extend the expressiveness of language by introducing equivalent terms. Here, the intent is to limit the variety of expression to a single consistent set of constructs based on equivalency of a variety of terms.

The foregoing and other features and advantages of the invention will be apparent from the following, more detailed description of preferred embodiments as illustrated in the accompanying drawings in which reference characters refer to the same parts throughout the various views.

FIG. 1 illustrates the process of finding and substituting synonyms for remote UI components according to the present invention.

FIG. 2 illustrates an example of hardware components of a control device within a home network that can be used to perform the present invention.

FIG. 3 illustrates a typical home network with a control CE device and slave CE devices whereto embodiments of the present invention can be applied.

It is to be understood by persons of ordinary skill in the art that the following descriptions are provided for purposes of illustration and not for limitation. An artisan understands that there are many variations that lie within the spirit of the invention and the scope of the appended claims. Unnecessary detail of known functions and operations may be omitted from the current description so as not to obscure the present invention.

A preferred embodiment of the present invention is illustrated in FIG. 1, in which one or more controlled devices can provide multiple remote user interface alternatives to the UI device 200. In order for a UI device to know which mechanisms are offered by a controlled device (and vice versa), it must be possible to discover this information. This discovery can be accomplished, for example, by means of a Remote I/O capability description which can be found in a similar way as service descriptions. After this information has been retrieved, a protocol can be chosen that is best suited for the situation.

One or more (remote) RUI definitions 101, which can be in any format, including RDP, X-Windows, VNC, HTTP, HAVi DDI, and UI Fragments, XForms and Java AWT, are taken (simultaneously) as input to an extraction process 102, where the relevant user interface data (i.e. assets) that makes up the user interface (such as labels, bitmaps, widgets) 120 are extracted from the incoming 110 UI definition. If the UI definition input 110 is defined on an abstract level, such as XForms, then this is accomplished in a relatively straightforward manner, as is known in the art. For less abstract or mixed input (such as RDP or HTML) known techniques such as model recovery [1] are applied to extract the relevant UI data 120 and the underlying semantic model of the UI from the incoming concrete UI 110.

After the extraction process 102 the user interface data is analyzed and transformed (syntactically or semantically) 103 using a database of "synonyms" 104:

<u>ORIGINAL</u>	<u>SYNONYM</u>
a, a', a'', a'''	a
b, b', b''	b'
x	x
y, y'	y'
...	...

If a match is found, and analysis of the user interface, with user preferences taken into consideration 107, shows that this replacement will indeed improve the user interface (e.g. by calculating a measure of confidence, and, optionally, through learning from previous analyses and transformations), then the preferred "synonym" replaces the original input 120 and the modified UI information 121 is output, which can be used to render the UI 106:

<u>UI</u>	<u>MODIFIED UI</u>
-----------	--------------------

label a'	label a
label b''	label b'
label c	label c
widget x	widget x
widget y	widget y'
...	...

If no match is found, the system connects to the Internet 105 to check on-line thesauri 109 or other relevant databases. If still no match is found, the original value remains in the UI.

The preferred "synonym" which is used for transformation can either be a default value (e.g. the first in the list) or one which is selected by a user 108. In a preferred embodiment, the user is allowed to add "synonyms", for example, to give names to devices and processes in a home network by inputting preferences 107. After this transformation step 103, the modified UI 106 is either shown to the user 108 or transferred to another device. In the first case, the synonym transformation process is done on a UI device. In the latter case, the synonym transformation process is offered as a service in the network.

Referring to FIG. 2, the control CE device within the home network of FIG. 3 may include a system with an architecture that is illustrated in the block diagram of FIG. 2. On power up, each slave CE device 101 transmits its RUI 110 to the control CE device 200 for a consistent presentation of its UI to a user by the control CE device 200. The control CE device 200 may include a transceiver 203, an extraction logic module 102, an analysis and transformation module 103, a synonym database 104, a memory 205 and a timer 204. The exemplary system 200 of FIG. 2 is for descriptive purposes only. Although the description may refer to terms commonly used in describing particular control CE devices, the description and concepts equally apply to other processing systems, including systems having architectures dissimilar to that shown in FIG. 2.

In operation, the transceiver 203 is coupled to an antenna (not shown) and to a wireline to transmit and receive data to and from wireless and wired slave CE devices 101.2 and 101.1, respectively. The extraction logic module 102 extracts components of a received slave device UI definition. After the extraction process 102 the extracted UI data 120 is analyzed and transformed by an analysis and transformation module 103 to replace components of the received slave device UI 120 with matching synonyms from the synonym database 104 to

produce a transformed UI definition 121, and is displayed on the user interface module 207 as the interface to the remote (slave) device or is transmitted by transceiver 203 to a second (or third, etc.) control device (not shown). The memory 205 is used by both the extraction logic module and the analysis and transformation module 103 to store learning derived from previous processing of slave CE device UIs and optionally update the synonym database 104 therewith. The timer 204 is used by the logic modules for time dependent processes, such as outdated transformed UIs for slave CE devices that have powered down.

FIG. 3 illustrates a representative home network whereto embodiments of the present invention are to be applied. As shown in FIG. 3, a control CE device 200 is coupled to a plurality of slave wired and wireless CE devices 101, which, through the wireline and wireless links of a home network 206 communicate their UIs to the control CE device 200. A key principle of the present invention is to provide a mechanism to substitute components of a consistent user interface for similar components in each slave CE device UI 101. The control CE device 200 performs an extraction of user interface components and then analyzes and transforms 103 them to finally yield a transformed UI 106 for each slave CE device 101. Although a limited number of CE devices is shown in FIG. 3 for illustrative purposes, it is to be understood that the home network can support communications between a much larger number of CE devices. Thus, the number of CE devices in FIG. 3 should not impose limitations on the scope of the invention. Further, the home network can be based on any of internet protocol (IP) (RFC 791), NETBEUI, Bluetooth, Zigbee, SCP, IEC61883, DVB and ATSC DTV.

In replacing components of the remote user interface, one or more control devices can be employ different replacement paradigms comprising predetermined similarity measures between components of the CUI and the RUI, predetermined improvement measures in the consistency of the RUI after replacement of an RUI component by a CUI component (this can be a cumulative measure depending on the number of such replacements or mappings), predetermined difference measures between components of the CUI and the RUI.

In an alternative embodiment, the control device also has one or more local UIs (LUIs), i.e. a user interface for a local application (see Glossary term "Application"), that it can display, in addition to displaying RUIs. Since these LUIs may contain some user interface constructs (components), which are similar to constructs (components) used in an incoming

RUI, it is useful to make the incoming RUIs not only consistent with each other but also with LUIs (so if the label "Play from DVR" is the preferred synonym, then this label has to be used in the RUIs as well as the LUIs). In a preferred embodiment, the synonym that is chosen in the transformation process 103 is the synonym that is used in an LUI, in order to make the incoming RUIs more consistent with LUIs of a control device.

While the preferred embodiments of the present invention have been illustrated and described, it will be understood by those skilled in the art that various changes and modifications may be made, and equivalents may be substituted for elements thereof without departing from the true scope of the present invention. In addition, many modifications may be made to adapt to a particular situation and the teaching of the present invention can be adapted in ways that are equivalent without departing from its central scope. Therefore it is intended that the present invention not be limited to the particular embodiment disclosed as the best mode contemplated for carrying out the present invention, but that the present invention include all embodiments falling within the scope of the appended claims.